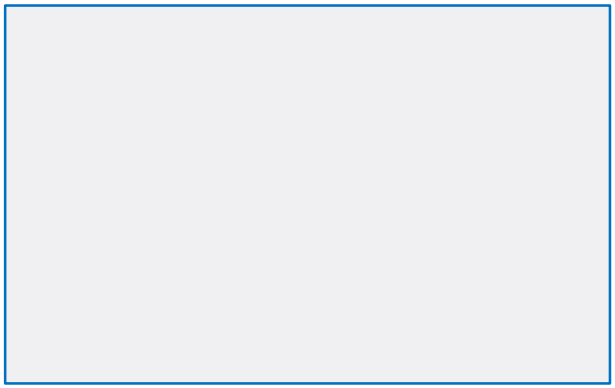
Text Files

Writing a Text File

```
int main()
{
   ofstream outFile( "grades.txt", ios::out );

   if( !outFile )
   {
      cout << "File could not be opened\n";
      system( "pause" );
      exit( 1 );
   }

   string id[] = { "1131411", "1131430", "1131432", "1131504" };
   string name[] = { "Dora", "Ariel", "Eason", "Brian" };
   int grade[] = { 91, 91, 83, 84 };

   for( int i = 0; i < 4; i++ )
      outFile << id[ i ] << ' ' << name[ i ] << ' ' << grade[ i ]
              << endl;
}
```

Writing a Text File

```
string id[] = { "1131411", "1131430", "1131432", "1131504" };
string name[] = { "Dora", "Ariel", "Eason", "Brian" };
int grade[] = { 91, 91, 83, 84 };
for( int i = 0; i < 4; i++ )
   outFile << id[ i ] << ' ' << name[ i ] << ' ' << grade[ i ] << endl;
```



grades.txt

	id	name	grade
0	1131411	Dora	91
1	1131430	Ariel	91
2	1131432	Dawn	83
3	1131504	Hsien	84

Writing a Text File

```
string id[] = { "1131411", "1131430", "1131432", "1131504" };
string name[] = { "Dora", "Ariel", "Eason", "Brian" };
int grade[] = { 91, 91, 83, 84 };
for( int i = 0; i < 4; i++ )
   outFile << id[ i ] << ' ' << name[ i ] << ' ' << grade[ i ] << endl;
```

```
1131411 Dora 91
```

grades.txt

	id	name	grade
0	1131411	Dora	91
1	1131430	Ariel	91
2	1131432	Dawn	83
3	1131504	Hsien	84

Writing a Text File

```
string id[] = { "1131411", "1131430", "1131432", "1131504" };
string name[] = { "Dora", "Ariel", "Eason", "Brian" };
int grade[] = { 91, 91, 83, 84 };
for( int i = 0; i < 4; i++ )
   outFile << id[ i ] << ' ' << name[ i ] << ' ' << grade[ i ] << endl;
```

```
1131411 Dora 91
1131430 Ariel 91
```

grades.txt

	id	name	grade
0	1131411	Dora	91
1	1131430	Ariel	91
2	1131432	Dawn	83
3	1131504	Hsien	84

Writing a Text File

```
string id[] = { "1131411", "1131430", "1131432", "1131504" };
string name[] = { "Dora", "Ariel", "Eason", "Brian" };
int grade[] = { 91, 91, 83, 84 };
for( int i = 0; i < 4; i++ )
   outFile << id[ i ] << ' ' << name[ i ] << ' ' << grade[ i ] << endl;
```

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
```

grades.txt

	id	name	grade
0	1131411	Dora	91
1	1131430	Ariel	91
2	1131432	Dawn	83
3	1131504	Hsien	84

Writing a Text File

```
string id[] = { "1131411", "1131430", "1131432", "1131504" };
string name[] = { "Dora", "Ariel", "Eason", "Brian" };
int grade[] = { 91, 91, 83, 84 };
for( int i = 0; i < 4; i++ )
   outFile << id[ i ] << ' ' << name[ i ] << ' ' << grade[ i ] << endl;
```

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

grades.txt

	id	name	grade
0	1131411	Dora	91
1	1131430	Ariel	91
2	1131432	Dawn	83
3	1131504	Hsien	84

Writing a Text File

- `!outFile`
  - Returns `true` if `failbit` or `badbit` set.
  - Opened non-existent file for reading, wrong permissions

- `cin >> myVariable`
  - `false` when `failbit` or `badbit` set, otherwise `true`.
  - `failbit` set when end-of-file indicator (*ctrl-z*) is entered.

- `while( cin >> myVariable )`
  - Loops until end-of-file indicator (*ctrl-z*) is entered.

file stream classes costructors

```
fstream( const char* filename,
         ios_base::openmode mode = ios_base::in | ios_base::out );

fstream( const string &filename,
         ios_base::openmode mode = ios_base::in | ios_base::out );


ifstream( const char* filename,
          ios_base::openmode mode = ios_base::in );

ifstream( const string &filename,
          ios_base::openmode mode = ios_base::in );


ofstream( const char* filename,
          ios_base::openmode mode = ios_base::out );

ofstream( const string &filename,
          ios_base::openmode mode = ios_base::out );
```

Parameters

`filename`

A string representing the name of the file to be opened.
Specifics about its format and validity depend on the library implementation and running environment.

`mode`

Flags describing the requested input/output mode for the file.
This is an object of the bitmask member type openmode that consists of a combination of the following member constants:

member constant	stands for	access
`in` *	input	File open for reading: the *internal stream buffer* supports input operations.
`out`	output	File open for writing: the *internal stream buffer* supports output operations.
`binary`	binary	Operations are performed in binary mode rather than text.
`ate`	at end	The *output position* starts at the end of the file.
`app`	append	All output operations happen at the end of the file, appending to its existing contents.
`trunc`	truncate	Any contents that existed in the file before it is open are discarded.

These flags can be combined with the bitwise OR operator (`|`).

* `in` is always set for `ifstream` objects (even if not set in argument *mode*).

Note that even though `ifstream` is an input stream, its internal `filebuf` object may be set to also support output operations.

Read One Record

```
int main()
{
   ifstream inFile( "grades.txt", ios::in );

   if( !inFile )
   {
      cout << "File could not be opened" << endl;
      system( "pause" );
      exit( 1 );
   }

   char id[ 8 ];   // student's id
   char name[ 8 ]; // student's name
   int grade;   // student's grade

   inFile >> id >> name >> grade;
   cout << left << setw( 9 ) << id << setw( 8 ) << name
        << right << setw( 5 ) << grade << endl;
}
```

Read All Records

```
int main()
{
   ifstream inFile( "grades.txt", ios::in );

   if( !inFile )
   {
      cout << "File could not be opened" << endl;
      system( "pause" );
      exit( 1 );
   }

   string id;   // student's id
   string name; // student's name
   int grade;   // student's grade

   cout << left << setw( 9 ) << "Id" << setw( 8 )
        << "Name" << "Grade" << endl;

   while( inFile >> id >> name >> grade )
      cout << left << setw( 9 ) << id << setw( 8 ) << name
           << right << setw( 5 ) << grade << endl;
}
```

```
int main()
{
   ifstream inFile( "grades.txt", ios::in );

   if( !inFile )
   {
      cout << "File could not be opened" << endl;
      system( "pause" );
      exit( 1 );
   }

   string id;   // student's id
   string name; // student's name
   int grade;   // student's grade

   inFile >> id >> name >> grade;
   while( !inFile.eof() )
   {
      cout << left << setw( 9 ) << id << setw( 8 ) << name
           << right << setw( 5 ) << grade << endl;
      inFile >> id >> name >> grade;
   }
}
```

```
while( inFile >> id >> name >> grade )
   cout << left << setw( 9 ) << id << setw( 8 ) << name
        << right << setw( 5 ) << grade << endl;
```

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

grades.txt

id	1131411
name	Dora
grade	91

Read All Records

```
Id        Name     Grade
```

Output

```
while( inFile >> id >> name >> grade )
   cout << left << setw( 9 ) << id << setw( 8 ) << name
        << right << setw( 5 ) << grade << endl;
```

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

grades.txt

id	1131411
name	Dora
grade	91

Read All Records

```
Id       Name     Grade
1131411  Dora        91
```

Output

```
while( inFile >> id >> name >> grade )
   cout << left << setw( 9 ) << id << setw( 8 ) << name
        << right << setw( 5 ) << grade << endl;
```

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

grades.txt

id	1131430
name	Ariel
grade	91

Read All Records

```
Id        Name     Grade
1131411   Dora        91
```

Output

```
while( inFile >> id >> name >> grade )
   cout << left << setw( 9 ) << id << setw( 8 ) << name
        << right << setw( 5 ) << grade << endl;
```

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

grades.txt

id	1131430
name	Ariel
grade	91

Read All Records

```
Id       Name    Grade
1131411  Dora       91
1131430  Ariel      91
```

Output

```
while( inFile >> id >> name >> grade )
   cout << left << setw( 9 ) << id << setw( 8 ) << name
        << right << setw( 5 ) << grade << endl;
```

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

grades.txt

id	1131432
name	Eason
grade	83

Read All Records

```
Id        Name     Grade
1131411   Dora        91
1131430   Ariel       91
```

Output

```
while( inFile >> id >> name >> grade )
   cout << left << setw( 9 ) << id << setw( 8 ) << name
        << right << setw( 5 ) << grade << endl;
```

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

grades.txt

id	1131432
name	Eason
grade	83

Read All Records

```
Id       Name    Grade
1131411  Dora       91
1131430  Ariel      91
1131432  Eason       83
```

Output

```
while( inFile >> id >> name >> grade )
   cout << left << setw( 9 ) << id << setw( 8 ) << name
        << right << setw( 5 ) << grade << endl;
```

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

grades.txt

id	1131504
name	Brian
grade	84

Read All Records

```
Id       Name    Grade
1131411  Dora       91
1131430  Ariel      91
1131432  Eason      83
```

Output

```
while( inFile >> id >> name >> grade )
   cout << left << setw( 9 ) << id << setw( 8 ) << name
        << right << setw( 5 ) << grade << endl;
```

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

grades.txt

id	1131504
name	Brian
grade	84

Read All Records

```
Id       Name    Grade
1131411  Dora       91
1131430  Ariel      91
1131432  Eason      83
1131504  Brian      84
```

Output

Open a Text File

```
ifstream inFile( "grades.txt", ios::in );
```

Open an existing file; if the file doesn't exit, fail.

```
ofstream outFile( "grades.txt", ios::out );
```

Create a file; if the file have existed, the data in it will be delete

Open a Text File

```
ifstream inFile( "grades.txt", ios::in );
ifstream inFile;
inFile.open( "grades.txt", ios::in );

ofstream outFile( "grades.txt", ios::out );
ofstream outFile;
outFile.open( "grades.txt", ios::out );
```

File Opening Errors

- Some possible file opening errors are:
  - attempting to open a nonexistent file for reading,
  - attempting to open a file for reading or writing without permission and
  - opening a file for writing when no disk space is available.

File Position

```
int main()
{
   ifstream inFile( "grades.txt", ios::in );

   if( !inFile )
   {
      cout << "File could not be opened" << endl;
      system( "pause" );
      exit( 1 );
   }

   string id;   // student's id
   string name; // student's name
   int grade;   // student's grade

   cout << left << setw( 9 ) << "Id" << setw( 8 )
        << "Name" << "Grade" << endl;

   while( inFile >> id >> name >> grade )
      cout << left << setw( 9 ) << id << setw( 8 ) << name
           << right << setw( 5 ) << grade << endl;
}
```

```
while( inFile >> id >> name >> grade )
   cout << left << setw( 9 ) << id
        << setw( 8 ) << name
        << right << setw( 5 ) << grade
        << endl;
```

file position 0

0	00110001	49	1
1	00110001	49	1
2	00110011	51	3
3	00110001	49	1
4	00110100	52	4
5	00110001	49	1
6	00110001	49	1
7	00100000	32	
8	01000100	68	D
9	01101111	111	o
10	01110010	114	r
11	01100001	97	a
12	00100000	32	
13	00111001	57	9
14	00110001	49	1
15			

grades.txt

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

```
while( inFile >> id >> name >> grade )
   cout << left << setw( 9 ) << id
        << setw( 8 ) << name
        << right << setw( 5 ) << grade
        << endl;
```

file position 8

0	00110001	49	1
1	00110001	49	1
2	00110011	51	3
3	00110001	49	1
4	00110100	52	4
5	00110001	49	1
6	00110001	49	1
7	00100000	32	
8	01000100	68	D
9	01101111	111	o
10	01110010	114	r
11	01100001	97	a
12	00100000	32	
13	00111001	57	9
14	00110001	49	1
15			

grades.txt

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

```
while( inFile >> id >> name >> grade )
   cout << left << setw( 9 ) << id
        << setw( 8 ) << name
        << right << setw( 5 ) << grade
        << endl;
```

file position 13

grades.txt

```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

0	00110001	49	1
1	00110001	49	1
2	00110011	51	3
3	00110001	49	1
4	00110100	52	4
5	00110001	49	1
6	00110001	49	1
7	00100000	32	
8	01000100	68	D
9	01101111	111	o
10	01110010	114	r
11	01100001	97	a
12	00100000	32	
13	00111001	57	9
14	00110001	49	1
15			

```
while( inFile >> id >> name >> grade )
   cout << left << setw( 9 ) << id
        << setw( 8 ) << name
        << right << setw( 5 ) << grade
        << endl;
```

file position 15

grades.txt

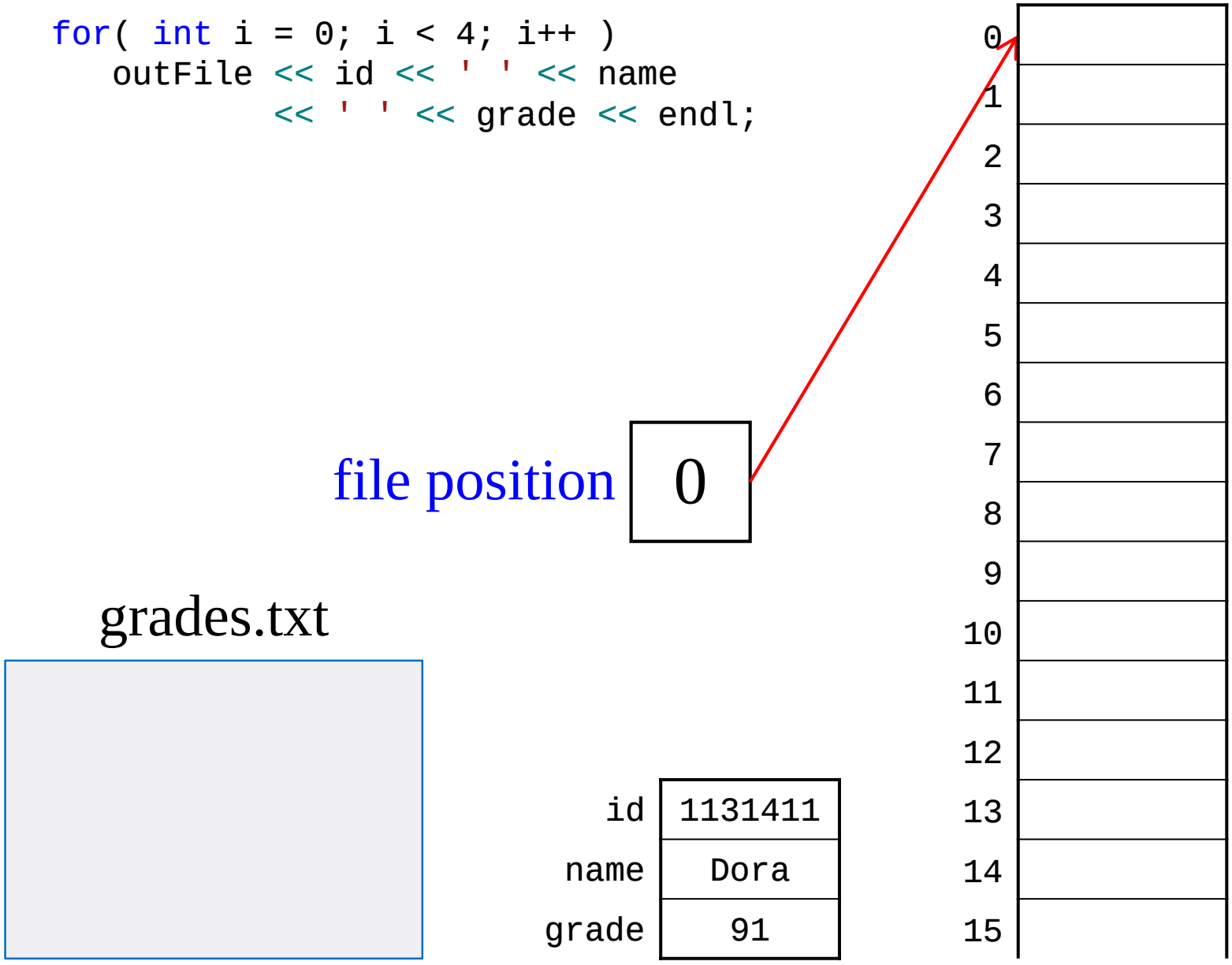
```
1131411 Dora 91
1131430 Ariel 91
1131432 Eason 83
1131504 Brian 84
```

0	00110001	49	1
1	00110001	49	1
2	00110011	51	3
3	00110001	49	1
4	00110100	52	4
5	00110001	49	1
6	00110001	49	1
7	00100000	32	
8	01000100	68	D
9	01101111	111	o
10	01110010	114	r
11	01100001	97	a
12	00100000	32	
13	00111001	57	9
14	00110001	49	1
15			

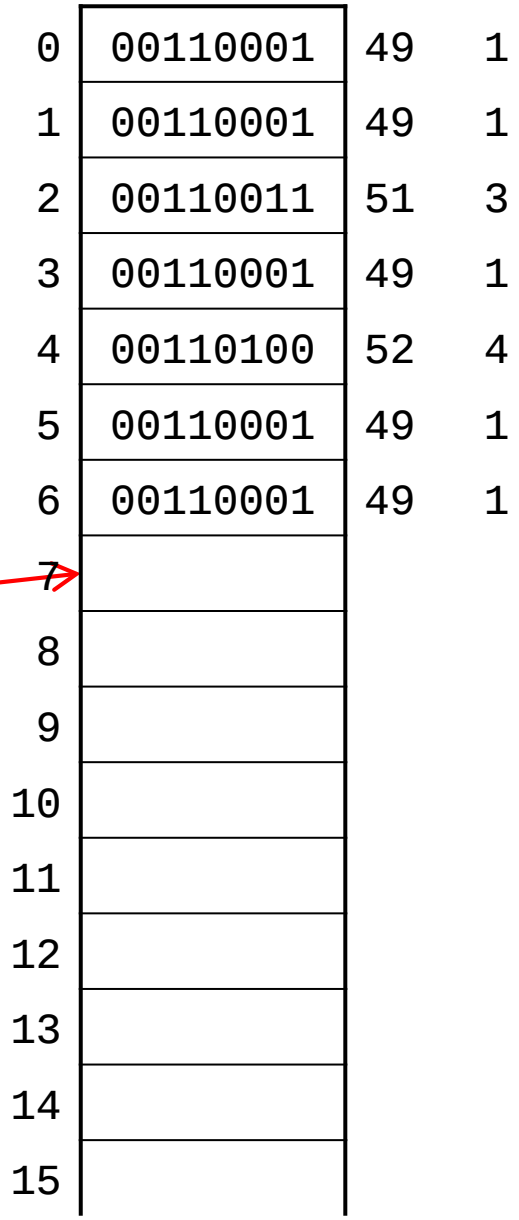
Writing a Text File

```
int main()
{
   ofstream outFile( "grades.txt", ios::out );

   if( !outFile )
   {
      cout << "File could not be opened\n";
      system( "pause" );
      exit( 1 );
   }

   string id[] = { "1131411", "1131430", "1131432", "1131504" };
   string name[] = { "Dora", "Ariel", "Eason", "Brian" };
   int grade[] = { 91, 91, 83, 84 };

   for( int i = 0; i < 4; i++ )
      outFile << id[ i ] << ' ' << name[ i ] << ' ' << grade[ i ]
              << endl;
}
```

```
for( int i = 0; i < 4; i++ )
   outFile << id << ' ' << name
           << ' ' << grade << endl;
```



```
for( int i = 0; i < 4; i++ )
   outFile << id << ' ' << name
           << ' ' << grade << endl;
```



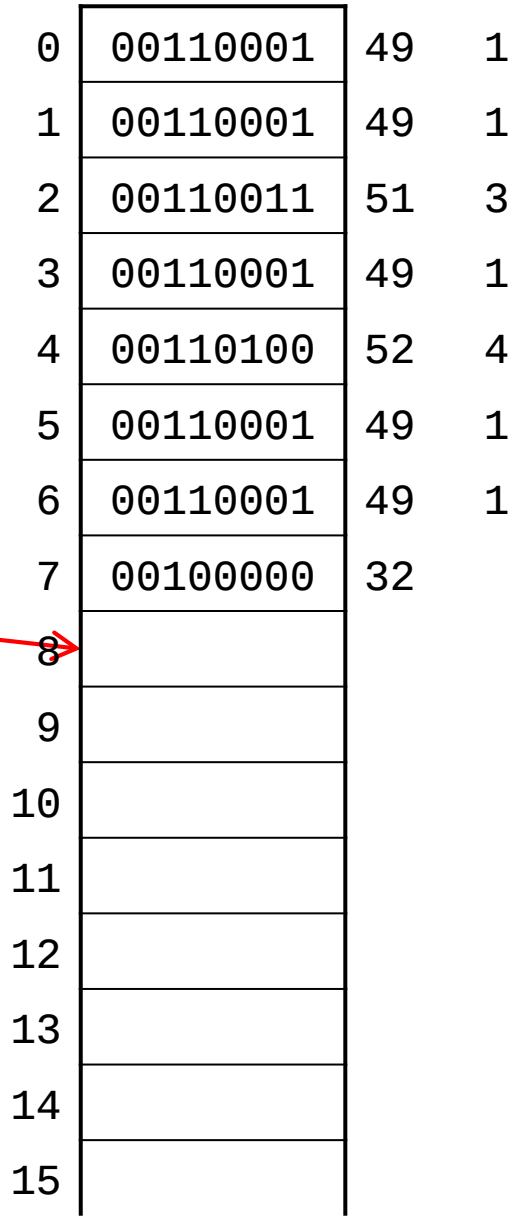
file position 7

grades.txt

```
1131411
```

id	1131411
name	Dora
grade	91

```
for( int i = 0; i < 4; i++ )
   outFile << id << ' ' << name
           << ' ' << grade << endl;
```



file position 8

grades.txt

```
1131411
```

id	1131411
name	Dora
grade	91

```
for( int i = 0; i < 4; i++ )
   outFile << id << ' ' << name
           << ' ' << grade << endl;
```

file position 12

grades.txt

```
1131411 Dora
```

id	1131411
name	Dora
grade	91

0	00110001	49	1
1	00110001	49	1
2	00110011	51	3
3	00110001	49	1
4	00110100	52	4
5	00110001	49	1
6	00110001	49	1
7	00100000	32	
8	01000100	68	D
9	01101111	111	o
10	01110010	114	r
11	01100001	97	a
12			
13			
14			
15			

```
for( int i = 0; i < 4; i++ )
   outFile << id << ' ' << name
           << ' ' << grade << endl;
```

file position 13

grades.txt

```
1131411 Dora
```

id	1131411
name	Dora
grade	91

0	00110001	49	1
1	00110001	49	1
2	00110011	51	3
3	00110001	49	1
4	00110100	52	4
5	00110001	49	1
6	00110001	49	1
7	00100000	32	
8	01000100	68	D
9	01101111	111	o
10	01110010	114	r
11	01100001	97	a
12	00100000	32	
13			
14			
15			

```
for( int i = 0; i < 4; i++ )
   outFile << id << ' ' << name
           << ' ' << grade << endl;
```

file position 15

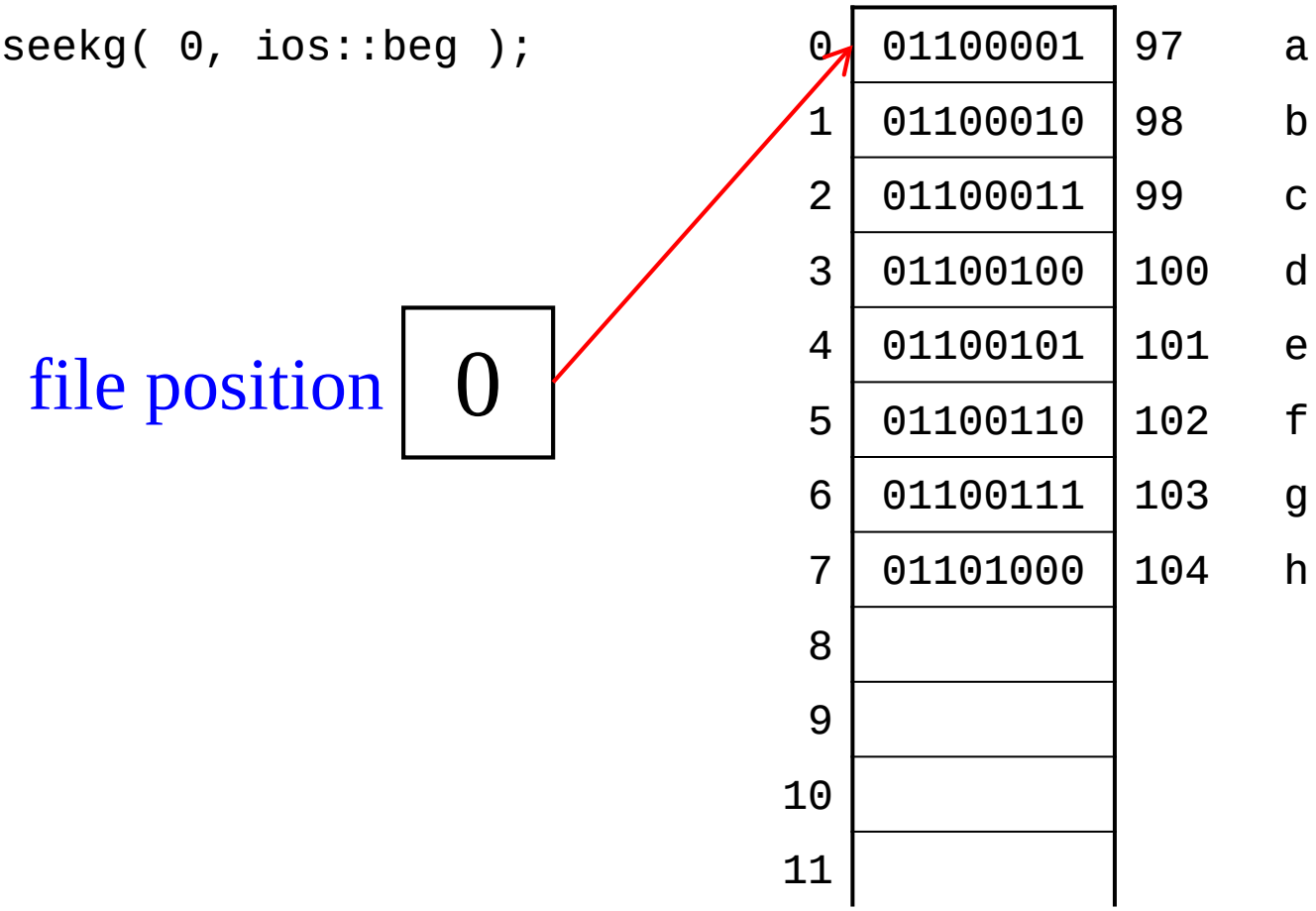
grades.txt

```
1131411 Dora 91
```

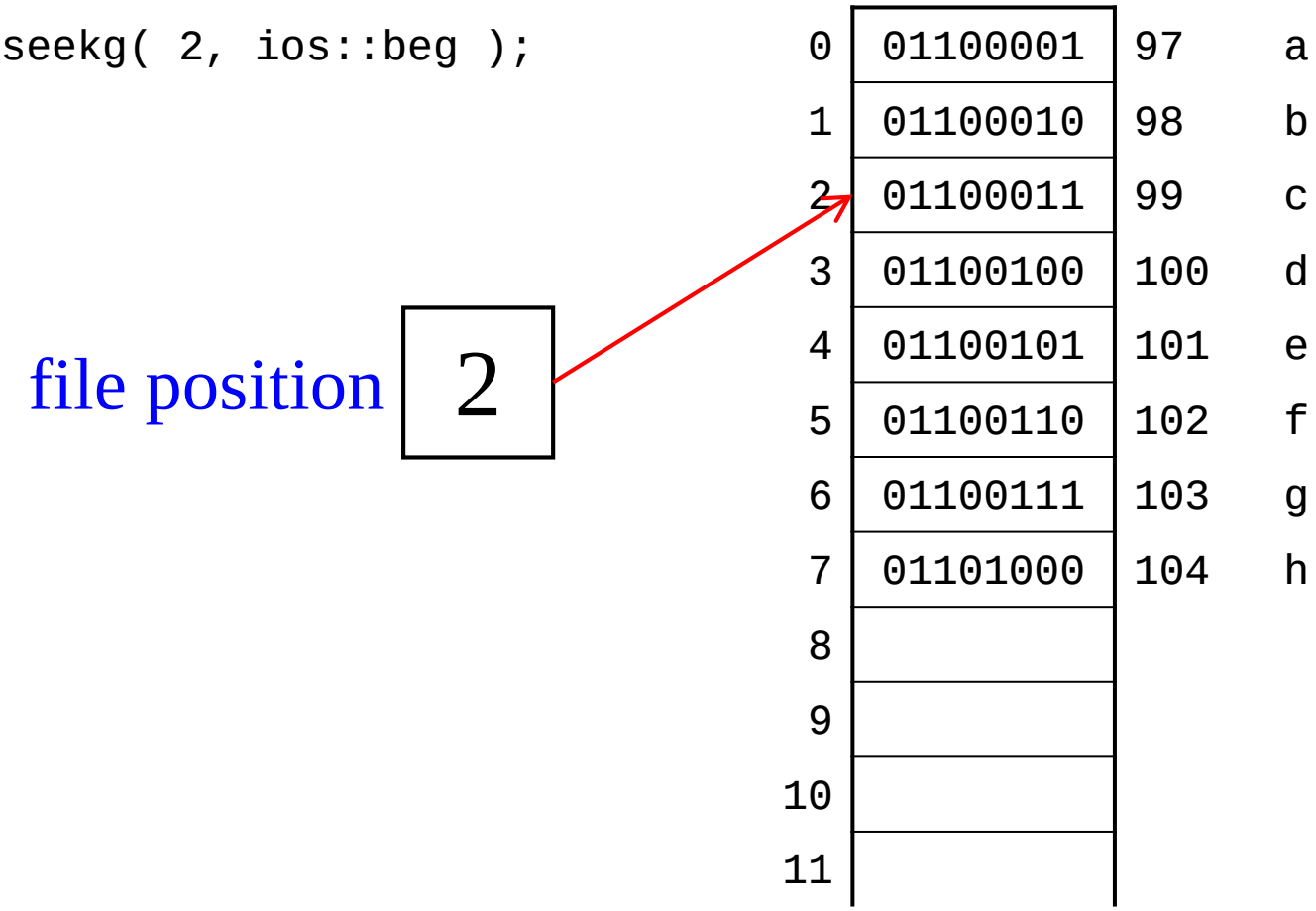
id	1131411
name	Dora
grade	91

0	00110001	49	1
1	00110001	49	1
2	00110011	51	3
3	00110001	49	1
4	00110100	52	4
5	00110001	49	1
6	00110001	49	1
7	00100000	32	
8	01000100	68	D
9	01101111	111	o
10	01110010	114	r
11	01100001	97	a
12	00100000	32	
13	00111001	57	9
14	00110001	49	1
15			

```
inFile.seekg( 0, ios::beg );
```



```
inFile.seekg( 2, ios::beg );
```



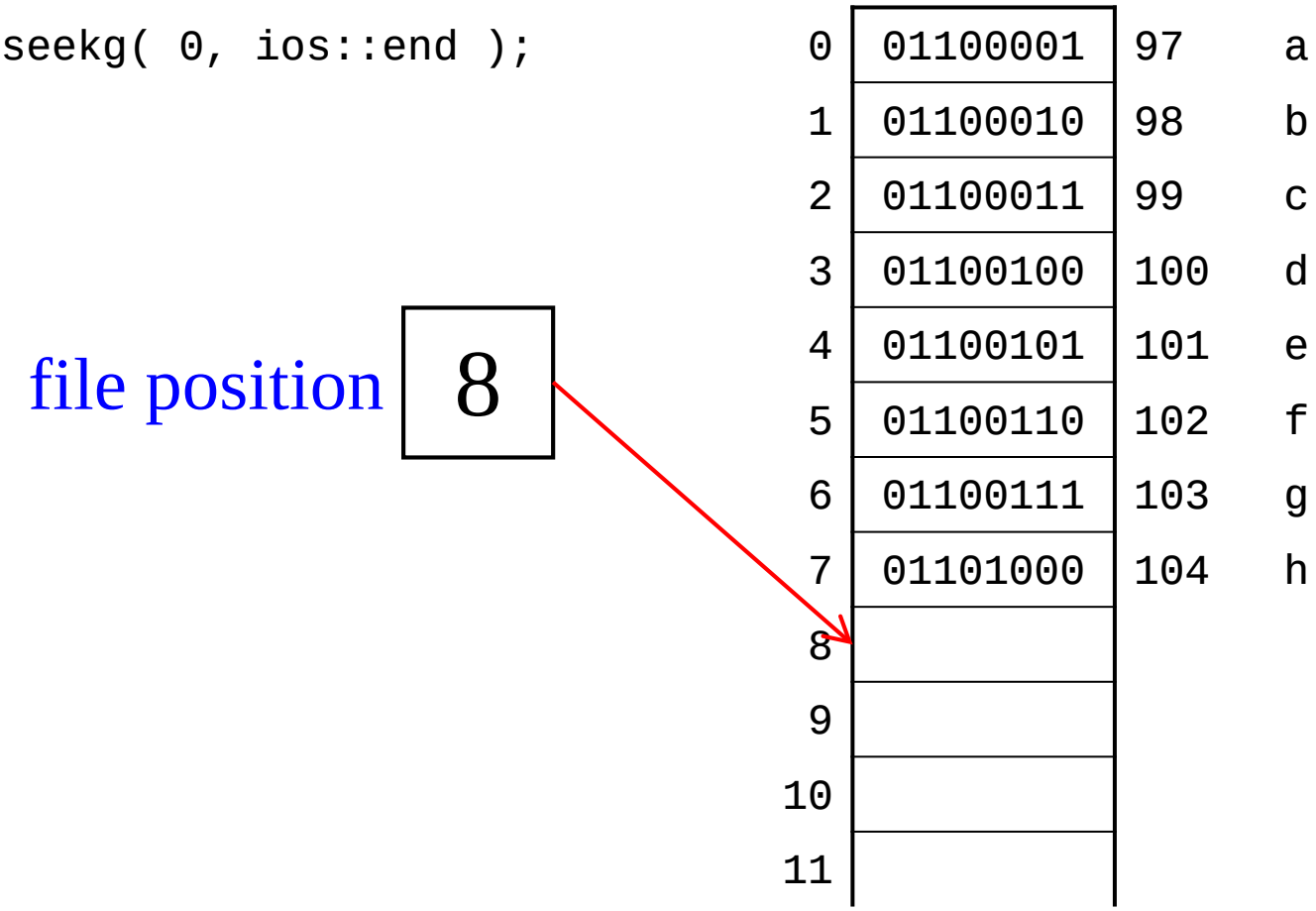
```
inFile.seekg( -1, ios::beg );
```

0	01100001	97	a
1	01100010	98	b
2	01100011	99	c
3	01100100	100	d
4	01100101	101	e
5	01100110	102	f
6	01100111	103	g
7	01101000	104	h
8			
9			
10			
11			

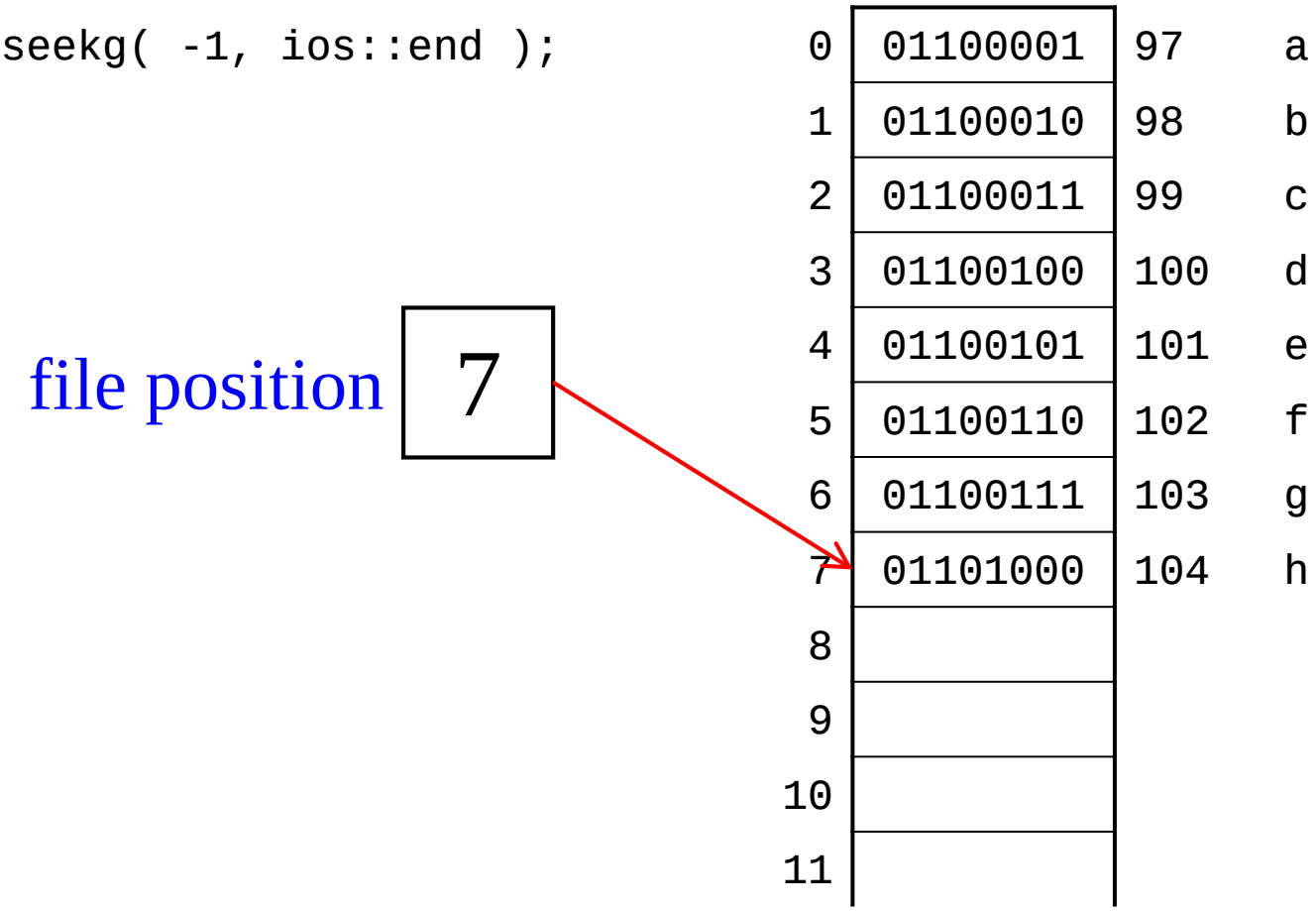
`inFile.seekg( -1, ios::beg )`

sets the `failbit` and resets the `goodbit`.

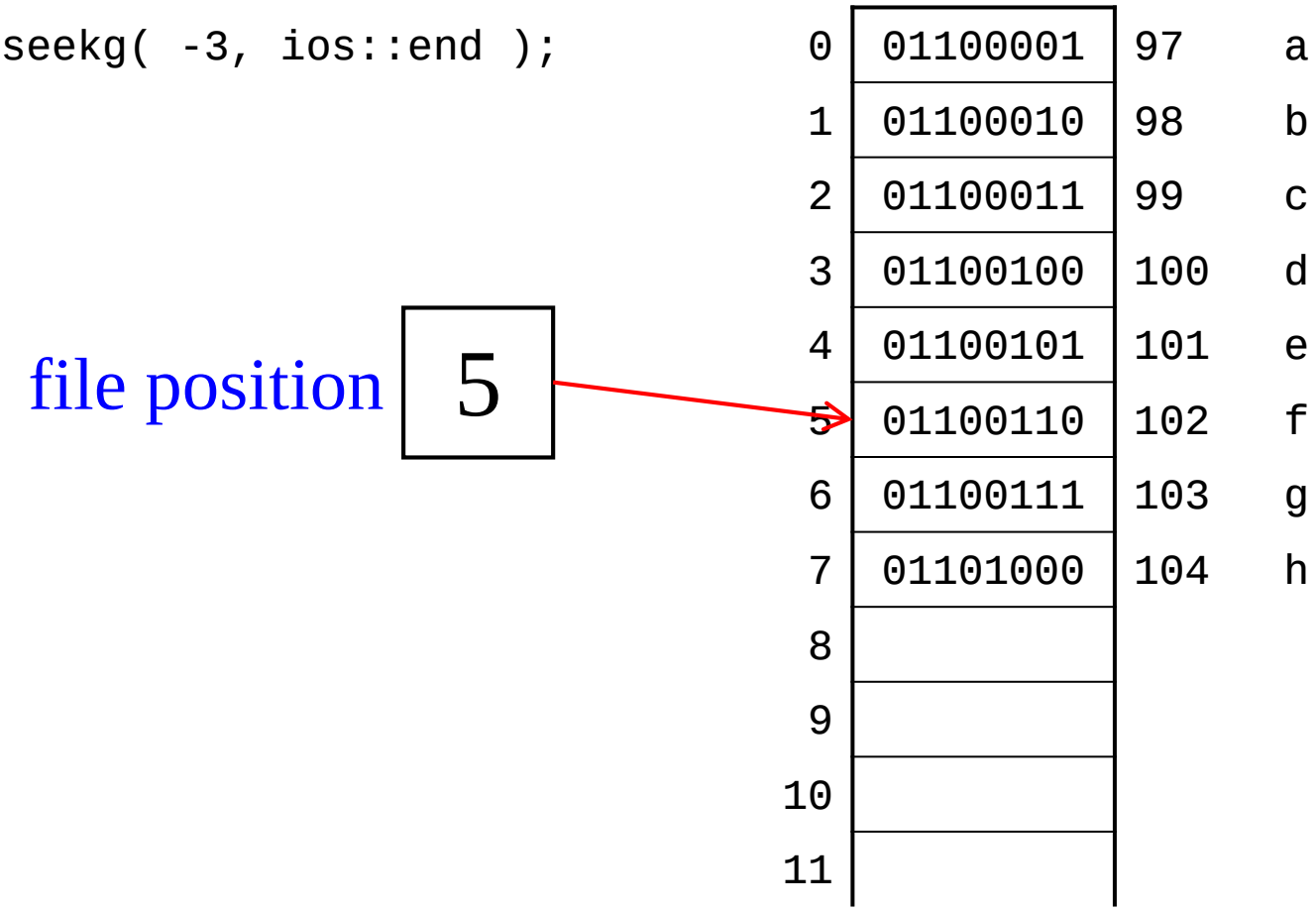
```
inFile.seekg( 0, ios::end );
```



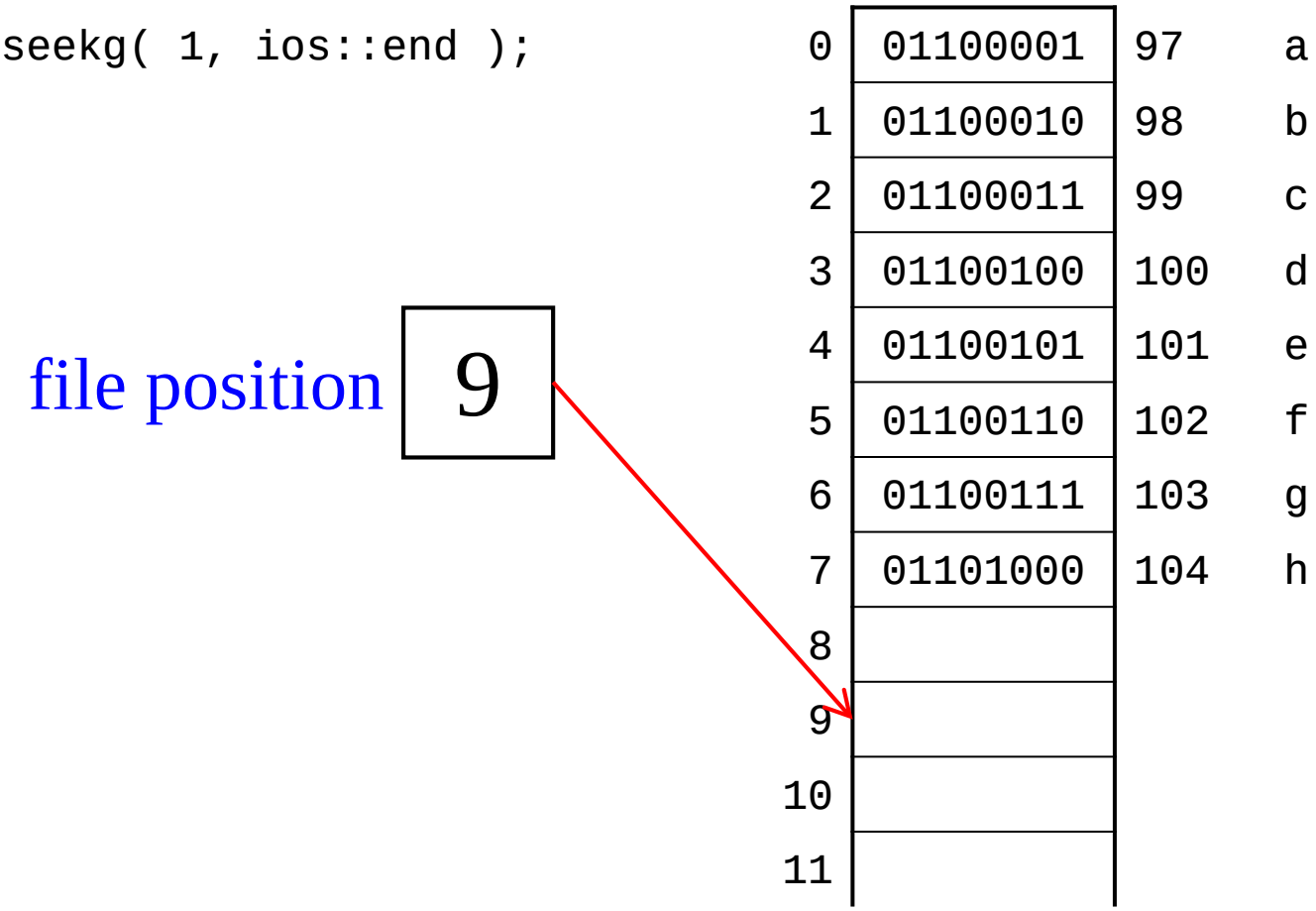
```
inFile.seekg( -1, ios::end );
```



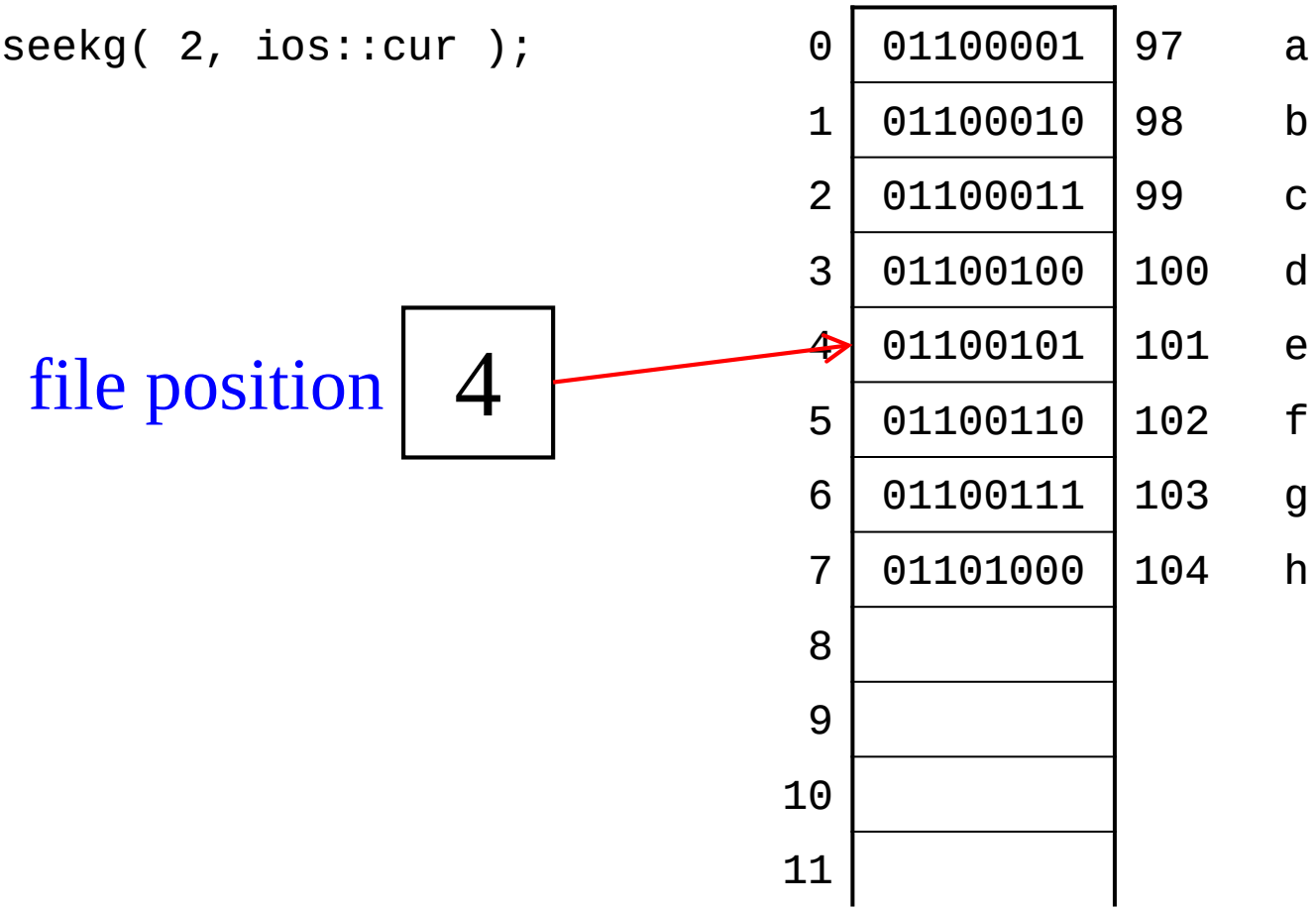
```
inFile.seekg( -3, ios::end );
```



```
inFile.seekg( 1, ios::end );
```

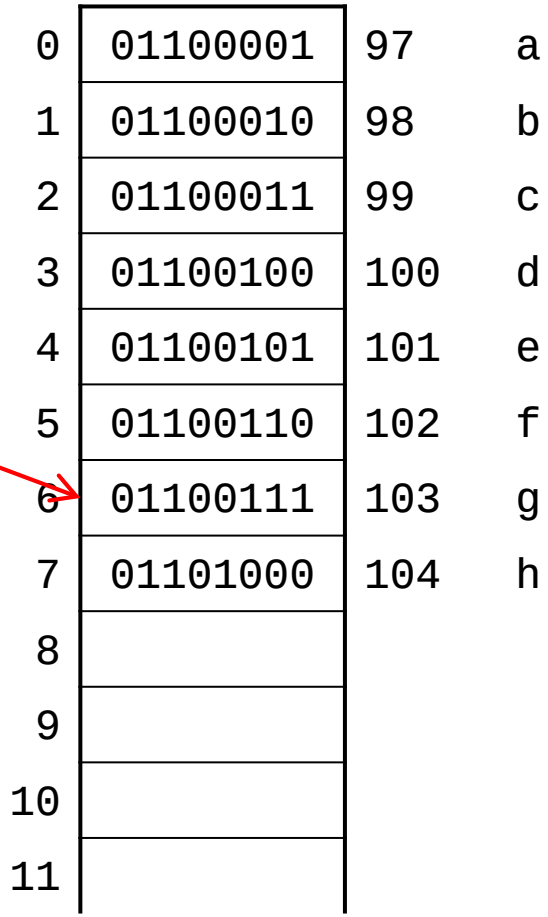


```
inFile.seekg( 2, ios::cur );
```

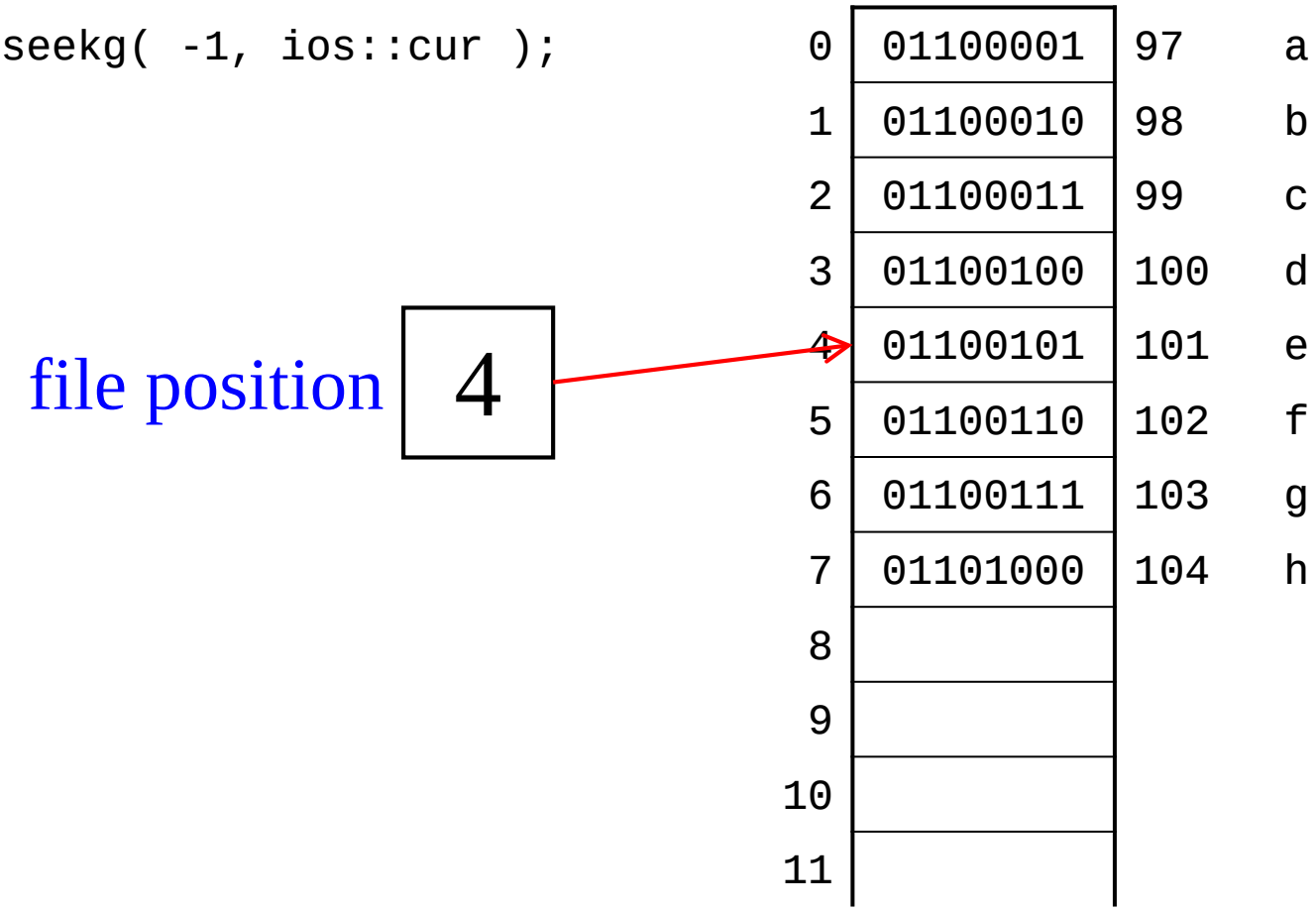


```
inFile.seekg( 2, ios::cur );
```

file position 6



```
inFile.seekg( -1, ios::cur );
```



```
inFile.seekg( -1, ios::cur );
```

file position 3